

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1-25 (canceled)

26. (new) A parser accelerator apparatus having an adaptive learning capability, the apparatus comprising:

- a finite state machine configured to parse a document;
 - a memory configured to store at least one state table;
 - a token buffer configured to store at least one token received from the finite state machine;
 - a parser accelerator compiler configured to compile a grammar specification and to generate a state transition specification in a self describing format; and
 - a parser accelerator loader configured to load the memory with a state table corresponding to the state transition specification,
- wherein the parser accelerator compiler and the parser accelerator loader are configured to reconfigure the parser accelerator in response to detected data patterns in an electronic document in substantially real time, thereby adapting over time to provide a learning capability.

27. (new) A method for dynamically reconfiguring a parser accelerator comprising:

- providing a grammar specification;

providing a parser accelerator having a finite state machine and a state table memory;

compiling the grammar to generate finite automata;

creating a finite state machine state transition specification in a self describing format from the finite automata; and

loading the finite state machine state transition specification into the state table memory.

28. (new) The method of claim 2, wherein the self-describing format is a mark-up language.

29. (new) The method of claim 3, wherein the mark-up language is extensible mark-up language (XML).

30. (new) The method of claim 2, wherein the grammar specification includes a specification of desirable performance characteristics of the parser accelerator.

31. (new) A parser accelerator apparatus having an adaptive learning capability, the apparatus comprising:

a finite state machine configured to parse a document;

a memory configured to store at least one state table;

a parser accelerator compiler configured to compile a grammar specification and to generate a state transition specification in a self describing format comprising a mark-up language; and

a parser accelerator loader configured to load the memory with a state table corresponding to the state transition specification,

wherein the parser accelerator compiler and the parser accelerator loader are configured to reconfigure the parser accelerator in response to a changed condition.

32. (new) The apparatus of claim 6 wherein the changed condition comprises a pattern in a document.

33. (new) A method for dynamically reconfiguring a parser accelerator comprising:

electronically providing a grammar specification including a specification of desirable performance characteristics of the parser accelerator;

providing a parser accelerator having a finite state machine and a state table memory;

compiling the grammar to generate finite automata;

creating a finite state machine state transition specification from the finite automata; and

loading the finite state machine state transition specification into the state table memory.

34. (new) The method of claim 8 wherein creating a finite state machine state transition specification from the finite automata comprises creating a finite state machine state transition specification in extensible mark-up language (XML).